

Therefore, the invention should be limited only by the spirit and scope of the appended claims.

- Sub a2
1. A method of testing a computerized application under test that allows simultaneous users over a computer network, the method comprising the steps of:
- providing test code that exercises a component of the application under test;
 - synchronizing and executing a plurality of instances of the test code and recording performance data on the component of the application under test;
 - repeating step b) multiple times, with a different number of instances of the test code;
 - analyzing the recorded performance data to indicate a performance characteristic of the component of the application under test in response to load.

Sub B2

2. The method of claim 1 wherein the step of providing test code includes generating test code automatically.

Sub a3

3. The method of claim 1 wherein the application under test is an object oriented language and the step of providing test code comprises providing test code to exercise one object in the application.

Sub B2

4. The method of claim 1 wherein the step of synchronizing comprises starting each instance of the test code at the same time.

5. The method of claim 1 wherein the step of synchronizing and executing comprises executing a portion of the plurality of instances of test code on a first computer and a portion of the plurality of instance of test code on a second computer connected to the network.

6. The method of claim 1 wherein the step of analyzing includes preparing a graphical display having as an independent variable the number of instances of the test code and the dependent variable is the performance data.

Sub B² 7.

1 The method of claim 1 wherein the step of analyzing includes preparing a
2 graphical display having as an independent variable the number of instances of the
3 test code and the dependent variable is derived from the performance data.
4

Sub 94
1 8.

2 The method of claim 1 wherein the application under test is resident on a first
3 server within the network and the application has a remote interface and the test
4 code is resident on at least a second computer within the network and exercises
the application under test using the remote interface of the application under test.

Sub B² 9.

1 The method of claim 1 wherein the step of analyzing includes displaying the
2 analyzed data to a human user using a graphical user interface.

Sub 95
1 10.

2 A method of testing a computerized application under test that allows
3 simultaneous users over a computer network, the method comprising the steps of:
4 a) specifying test conditions through a user interface to a test system;
5 b) initiating through a user interface to the test system the gathering of test
6 data on the performance of a at least one component of the application
7 under test at a plurality of load conditions;
8 c) specifying through a user interface to the test system the output format of
9 the test data; and
10 d) displaying in the specified format the response of at least one component
of the application under test to load.

Sub B² 11.

1 The method of claim 10 wherein the specified format is a graphical format
2 indicating response time as a function of load conditions.

1 12. The method of claim 11 wherein the specified graphical format is a Hi-Lo plot.

1 13. The method of claim 11 wherein the step of gathering data under a plurality of
2 load conditions comprises initiating the execution of a plurality of copies of a test
3 program, with the number of copies executing simultaneously relates to the load
4 condition.

1 14. The method of claim 13 wherein the step of specifying an output format includes
2 specifying a method by which response is measured.

1 15. The method of claim 13 wherein the step of gathering test data includes recording
2 the execution time between selected points in the test program for each
3 simultaneously executing copy of the test program and analyzing the recorded
4 execution times for all copies of the test program.
5

1 16. The method of claim 15 wherein the step of analyzing comprises determining the
2 average and maximum execution times for each of the load conditions.

1 17. The method of claim 10 wherein:

- 2 a) the computerized application under test comprises software resident on a
- 3 server controlling access to a computerized database;
- 4 b) the server is connected to a network and the application under test is
- 5 simultaneously accessed by a plurality of clients over the network; and
- 6 c) the test system is resident on at least a second server connected to the
- 7 network.

1 18. A method of testing a computerized application under test that allows
2 simultaneous users over a computer network, the application under test having a
3 plurality of software components, the method comprising the steps of:
4 a) providing test code to exercise a component;
5 b) creating a first plurality of copies of the test code;
6 c) simultaneously executing the first plurality of copies of test code while
7 recording times between events in each of the first plurality of copies of
8 test code;
9 d) creating a second plurality of copies of test code,
10 e) simultaneously executing the second plurality of copies of test code while
11 recording times between events in each of the second plurality of copies of
12 test code;
13 f) repeating a predetermined number of times the steps of creating plural
14 copies of the test code and simultaneously executing the plural copies
15 while recording event times; and

- 16 g) analyzing the recorded times to present information on the performance of
17 the component of the application under test as a function of load.
- 1 19. The method of claim 18 wherein the components comprise enterprise Java beans.
- 1 20. The method of claim 19 wherein each component has a plurality of functions
2 therein and the test code exercises functions of the components.
- 1 21. The method of claim 20 wherein the events at which times are recorded includes
2 times at which commands are issued to access functions of the components and
3 times at which execution of the commands are completed.
- 1 22. A system for determining performance of an application under test in response to
2 load, the system comprising:
3 a) coordination software;
4 b) at least one code generator, receiving as an input commands from the
5 coordination software and having as an output client test code;
6 c) at least one test engine, receiving as an input commands from the
7 coordination software, the test engine comprising a computer server
8 having a plurality of threads thereon, each thread executing an instance of
9 the client test code;
10 d) at least one data log having computerized memory, the memory holding
11 timing data created by the instances of the client test code in the plurality
12 of threads; and
13 e) at least one data analyzer software, operatively connected to the data log,
14 having an output that represents performance of the application under test
15 in response to load.
16